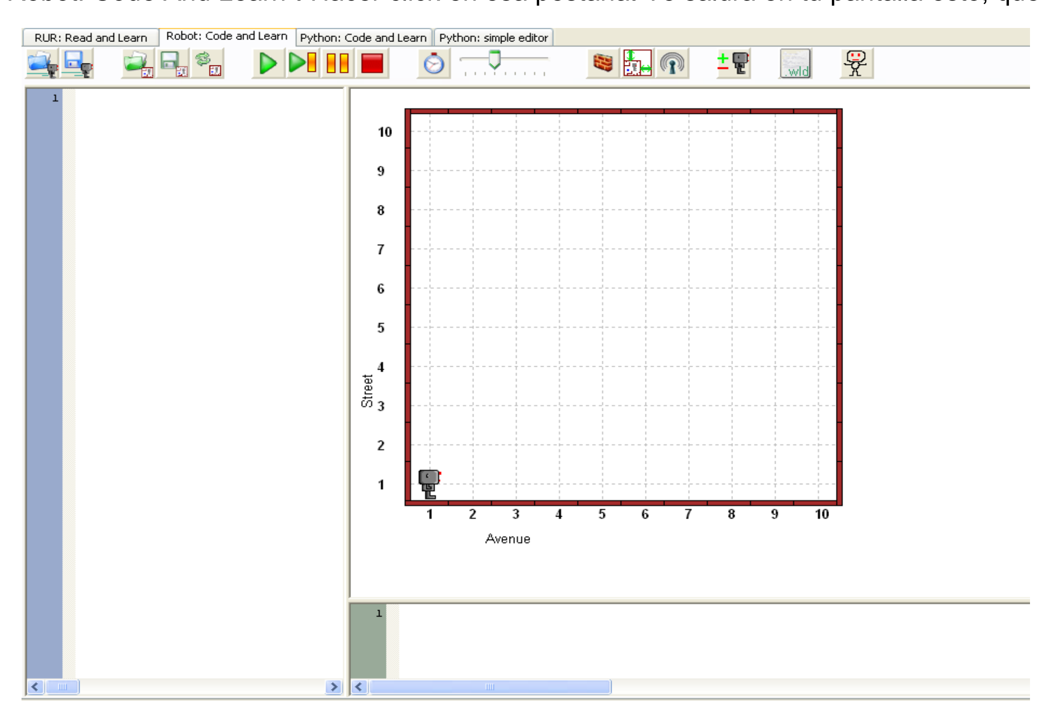
MANUAL DE USUARIO

CÓMO PROGRAMAR AL ROBOT

Para poder usar el programa RUR-PLE debes instalarlo en tu PC. Esto lo puedes hacer en la página siguiente: http://sourceforge.net/projects/rur-ple/files/. El programa RUR-PLE es un ambiente de aprendizaje del lenguaje de programación PHYTHON. Con él, aprenderás a realizar una programación sencilla primero, y después avanzarás de nivel hasta llegar a la parte compleja, que es el lenguaje PHYTHON.

Sigue los siguientes pasos para un básico aprendizaje de RURPLE:

1. Ya instalado tu programa, debes buscar la pestaña en la parte superior izquierda la que tiene el nombre de: “Robot: Code And Learn”. Hacer click en esa pestaña. Te saldrá en tu pantalla esto, que es un mundo que

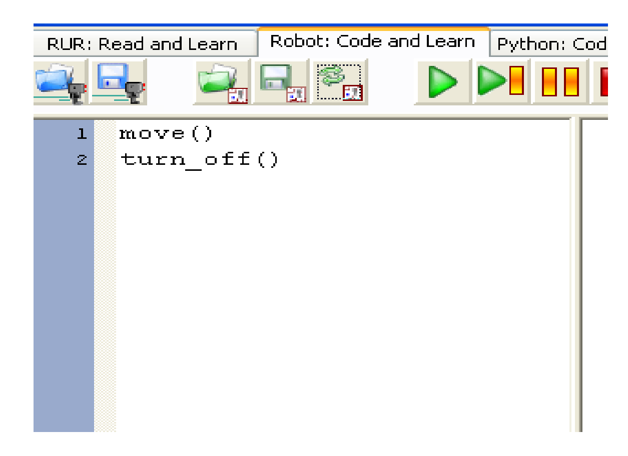


tiene calles y avenidas:

2. Para empezar a mover tu robot, puedes hacer click sobre él para despertarlo. Después puedes moverlo con la flechas de tu teclado.

Ahora a empezar a programar:

3. Para mover tu robot, no debes decirle a la computadora qué hacer en tu lenguaje, sino que debes escribirlo en el lenguaje de la computadora. Para que se mueva en una de las avenidas, puedes escribir el código “move()” en la parte izquierda de tu programa. Debes poner move() las veces que quieras que tu robot se mueva. Después de cada acción que le mandes a tu robot, debes decirle a tu robot que se apague, esto lo haces con el código “turn_off()”.



Para probar tu programación y ver como se mueve el robot, presiona el botón de „Play 

4. La computadora solo obedecerá comandos en inglés, pero tú puedes hacer que aprenda otro idioma. Debes poner, por ejemplo si tú quieres que en vez de move, tu robot aprenda avance:

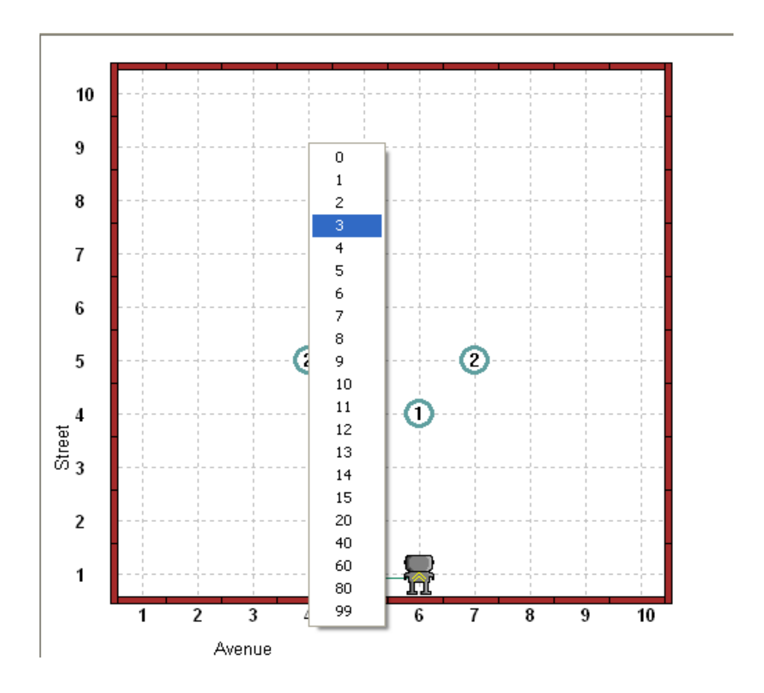
avance = move, entonces ya puedes darle la función de:

avance()

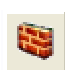
turn_off()

5. Para mover a tu robot hacia alguna dirección puedes utilizar: "turn_left ()"

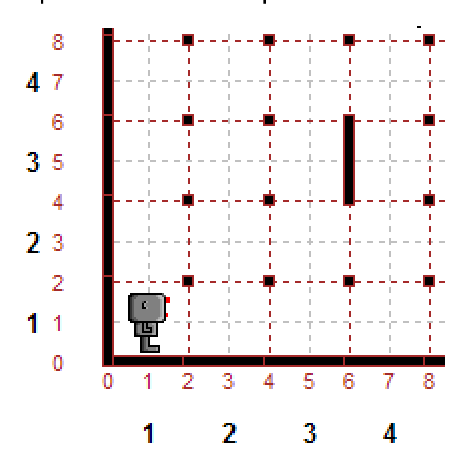
6. En el programa también puedes insertar "beepers" que son pequeños objetos en el mundo del robot. Para poder insertarlos, debes hacer click derecho en las esquinas de las calles y avenidas.



7. Para recoger estos beepers debes ingresar el código "pick_beeper()", y para poner un beeper debes ingresar el código de "put_beeper()".

8. También puedes construir paredes por medio del icono de  . Te saldrá algo como esto en tu pantalla:

Solo debes hacer click en las líneas punteadas para construir una pared como la de la imagen ejemplo.



9. Si quieres que el robot cruce a la derecha, podemos definirlo con 3 turn_left() para que puedas poner solo turn_right() en vez de los otros 3 códigos, por ejemplo:

```
def turn_right():
    turn_around()
turn_left()
```

Puedes definir lo que quieras para en el futuro ahorrarte escribir todas las funciones muchas veces. **10.** También existen repeticiones de alguna definición, por ejemplo:

```
def turn_right():
  repeat(turn_left, 3)
```

Hicimos que nuestra definición de turn_right() la realice 3 veces.

11. También hay otras funciones que nos puede ayudar al movimiento y programación del robot: while (significa mientras), if (significa si), else (significa sino), if not (se utiliza cuando no se cumple la condición establecida). Ejemplos:

```
if on_beeper():
  pick_beeper()

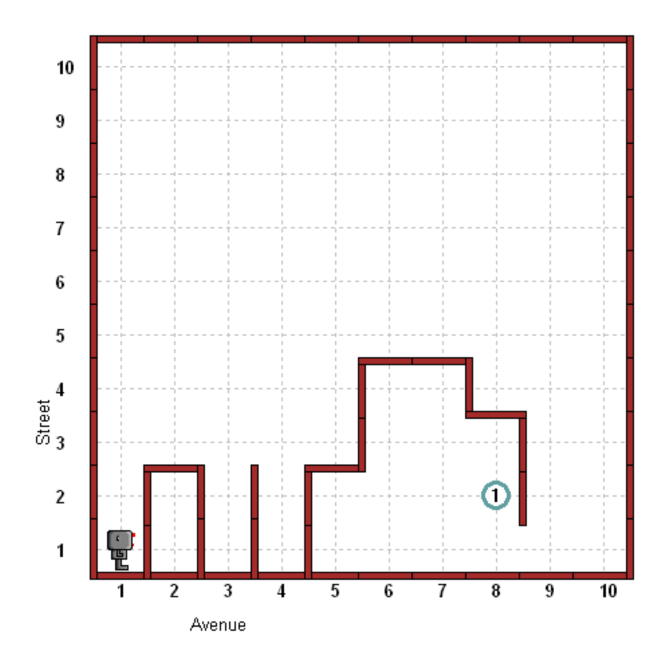
if front_is_clear():
  move()
else:
  turn_left()
turn_off()
```

```
while not on_beeper():
  move()
turn_off()
```

12. Si quieres agregar algún comentario a tu programación, solo debes poner el signo # antes de escribir el comentario.

Ahora a realizar un programa para nuestro robot.

Vamos a indicarle al robot a que pueda realizar cualquier acción dentro de un mundo que le será propuesto. Que su objetivo es va en busca de un beeper y donde lo encuentra se para pero para ello tiene que evitar paredes o muros hasta que lo encuentra. Puede visualizar la posición en que se encuentra el robot.



1. Para que el robot se ponga en movimiento utilizaremos la función ‘def right’.
2. Si en frente del robot hay un muro entonces utilizaremos la función ‘ while’, sino, también.
3. y el ‘if ‘si llega a encontrar un beeper.

En la parte inferior está ejemplificado en la forma en que está estructurado y la forma en que está hecho el programa.

Moverse hacia la Derecha.

Hacer mientras, avanzar hacia la izquierda.

Si al moverse no hay obstáculo se usa esta función, hacer mientras.

Cuando encuentra un beeper se apaga.

```
def right():
    repeat(turn_left,3)
def empieza_con_muro():
    while not front_is_clear():
        turn_left()
        while front_is_clear():
            move()
            if on_beeper():
                turn_off()
            elif right_is_clear():
                right()
def empieza_sin_muro():
    while front_is_clear():
        move()
        while not front_is_clear():
            if right_is_clear():
                right()
                move()
            else:
                turn_left()
        if on_beeper():
            turn_off()
        elif right_is_clear():
            right()
empieza_con_muro()
empieza_sin_muro()
turn_off()
```

Esta función se usa si al inicio hay muro.

Se apaga si encuentra un beeper.

Gira a la derecha si está libre.

Hacer mientras, si en su paso está libre gira a su derecha.

Sino que se mueva a su izquierda.

Finalmente, ya está el mundo del robot y el programa, click en ejecutar. El robot quedará en el siguiente mundo.

